A Scene Text-Based Image Retrieval System

Thuy Ho
Faculty of Information System
University of Information Technology
Ho Chi Minh city, Vietnam
thuyhtn@uit.edu.vn

Ngoc Ly
Faculty of Information Technology
University of Science
Ho Chi Minh city, Vietnam
lqngoc@fit.hcmus.edu.vn

***Abstract*— The rapidly increasing popularity of digital images raise a question of how to automatically index and help users navigate such libraries of images. One approach is to extract text appearing in images which often gives an indication of a scene's semantic content. However, it can be challenging since the text is often embedded in a complex background. This paper presents a novel approach for querying images using scene text. The text in natural scene images is localized and extracted properly by a novel mechanism. Then, an indexing and retrieval scheme is introduced. The proposed methods are evaluated on the public database of the ICDAR 2003 competition. Experimental results are very encouraging and suggest that these algorithms can be used in image retrieval applications.**

***Keywords*— text detection, text binarization, scene text, image retrieval, image indexing**

I. Introduction

With the widely use of digital image capture devices, such as digital cameras, mobile phones and PDAs, quantities of available images are rapidly increasing. This increasing availability has rekindled interest in the problem of how to index multimedia information sources automatically as well as how to browse and manipulate them efficiently. Traditionally, images have been manually annotated with a small number of keywords descriptors. Unfortunately, comparing the volume of image files available at the current time, it is difficult to extract information from these files manually through human intervention. This has motivated many researchers to design and develop new algorithms to index and store image on a database for faster retrieval. Among all contents in images, such as face, human, scene, etc., text is found to be one of the most important features to understand the image content and has been used to index and retrieve images. If the text in an image can be extracted, it can provide natural, meaningful keywords indicating the image's content.

As an essential prerequisite for scene text-based image search, text within images has to be robustly located. Traditional OCR based methods are not directly applicable to images of text. This is mainly because of the difficulties in detecting texts in images. The majority of OCR engines is designed for scanned text and so depends on segmentation which correctly separates text from background pixels. While this is usually simple for scanned text, it is much harder in natural images. Natural images are usually at lower resolution and have added complexities due to the variations of font, size, color, alignment, and lighting condition. In addition, text is often embedded in a complex background, light shadow, non-planar objects. Although some existing methods have reported promising results, there still remain several difficult problems need to be solved.

Most of the previous studies in text detection can be classified into approaches based on edge, connected component, and texture [1]. Edge based methods are focused on trying to find out regions on the image where there is a high contrast between text and background in order to detect and to merge edges from letters in images [2][3]. Edge based methods is fast and can have a high recall. However, it often produces many false alarms since the background may also have strong edges similar to the text. Texture based methods [4][5][6] classify text regions from non-text regions using texture features. The texture based methods mostly use texture analysis approaches such as Gaussian filtering, Wavelet decomposition, Fourier transform, Discrete Cosine Transform (DCT) or Gabor filtering in order to obtain texture information from images. Machine learning methods are often used in this approach. One problem of this approach is that some texture features are of high dimensions, adding to the complexity of classification. Connected components (CCs) based methods [7][8] use bottom-up approach which is based on iteration to merge and combine connected pixels by the help of homogeneity criterion. These methods normally include four steps: i) pre-processing, such as color clustering and noise reduction, ii) CC generation, iii) filtering out non-text components and iv) component grouping [1]. Compared with the texture based approaches, the CC based approaches have the advantages of quicker computation and less sensitive to font size. However, their performances drop dramatically when detecting texts in complex backgrounds.

Numerous reports have been published about indexing and retrieval of digital images, each concentrating on different aspects (see [9]). Automatic image indexing generally uses indices based on the color, texture, or shape of objects. Other systems are restricted to specific domains such as newscasts, or soccer. None of them tries to extract and recognize automatically the text appearing in digital images and use it as an index for retrieval.

In this paper, we present a new approach to image retrieval using scene text. In which, the text in natural scene images is detected and recognized by a novel method. We also demonstrate that the proposed methods enable semantic indexing and retrieval. Overall, our paper offers the following main contributions:

978-1-4673-5604-6/12/$31.00 ©2012 IEEE

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **DEC 2012**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**

4. TITLE AND SUBTITLE

A Scene Text-Based Image Retrieval System

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Faculty of Information System University of Information Technology Ho Chi Minh city, Vietnam

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR’S ACRONYM(S)

11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

See also ADA587934. IEEE International Symposium on Signal Processing and Information Technology (12th) (ISSPIT 2012) Held in Ho Chi Minh City, Vietnam on December 12-15, 2012. AOARD-CSP-131010.

14. ABSTRACT

The rapidly increasing popularity of digital images raise a question of how to automatically index and help users navigate such libraries of images. One approach is to extract text appearing in images which often gives an indication of a scene’s semantic content. However, it can be challenging since the text is often embedded in a complex background. This paper presents a novel approach for querying images using scene text. The text in natural scene images is localized and extracted properly by a novel mechanism. Then, an indexing and retrieval scheme is introduced. The proposed methods are evaluated on the public database of the ICDAR 2003 competition. Experimental results are very encouraging and suggest that these algorithms can be used in image retrieval applications.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**SAR**	**6**	

- We propose a new localization-verification scheme to robustly detect text strings with variations of font style, size, color and scale from complex natural scene images.
- Our system provides a reliable binarization for the detected text, which can be passed to OCR for text recognition.
- We propose a new approach to image retrieval using scene text. The our proposed model offers to support users to search for images even when it is unable to input keywords due to some cause (for instance not knowing the language of keywords).

The remainder of paper is organized as follows. Section II shows our approach overview. Section III provides a detailed description of our algorithms for text detection and recognition. In section IV, we introduce a indexing and retrieval scheme. Section V shows experimental results. In section VI, we draw conclusions.

II. System Overview

The overall process of the text detection and extraction is illustrated in Fig. 1. Our method is composed of two parts: text localization and text verification. The first part (text localization) consists of two steps: pre-processing and generation of candidate text regions. At pre-processing step, a procedure based on reconstruction transform is applied to remove noise as non-text regions. Next, edge features and morphological dilation are employed to locate image blocks. We apply Stroke Width Transform from [10] with some modification to generate candidate letters. These letters are paired to identify text lines, which are subsequently separated into words. At text verification stage, we use Histogram of Oriented Gradient (HOG) features to train a Support Vector Machines (SVM)-based classifier to determine whether a candidate word is text or not. Then the text regions are extracted by a novel binarization algorithm. The binarized text is recognized by OCR software. We apply a simple OCR post-correction model on the OCR output to improve the recognition performance. Finally, the recognized words are used as query keywords for retrieval.

III. Text Detection and Recognition

A. Pre-processing

Generally, the color of the text in an image is lighter or darker as compared to its background. For filtering out non-text regions, we apply reconstruction transformation on the gray scale image. In natural scene images, the information to be gathered should be within image not in the border regions of the image. As this step is used to remove objects connected to borders and lighter than it's surrounding. The reconstruction by dilation of a mask image g from a marker image f ($D_f = D_g$ and $f \le g$) is defined as the geodesic dilations of f with respect to g iterated until stability and is denoted by $R_g^{\delta}(f)$:

$$R_g^{\delta}(f) = \delta_g^{(i)}(f) \qquad (1)$$

where i is such that $\delta_g^{(i)}(f) = \delta_g^{(i+1)}(f)$ [11]. We implement the reconstruction based on algorithm described in [12]. The detail of the algorithm can be found in that paper.

According to Soille [11], the reconstruction transformation can be used to extract connected image objects having higher intensity values than the surrounding objects. The marker image equals then zero everywhere except the points **x** which have a value equal to that of the image g at the same position. In order to remove objects connected to the image border, we use the input image as mask image and the marker image zero everywhere except along its border where the values of the mask image are considered. The removal of objects connected to the image border is illustrated in Fig. 2. Then we apply a threshold in the reconstructed image to produce a binary one. After this processing step, many non-text regions are removed. We have seen that this process reduce noise and enhance the text regions of the image.

B. Generation of candidate text regions

To reduce noise and connect strokes in the binary image, we use two morphological operators: closing operation and dilation operation. In low contrast images, a character sometimes might be broken to pieces. Thus, it is more likely that connected component analysis might make the wrong decision. So, we have to merge these regions first.

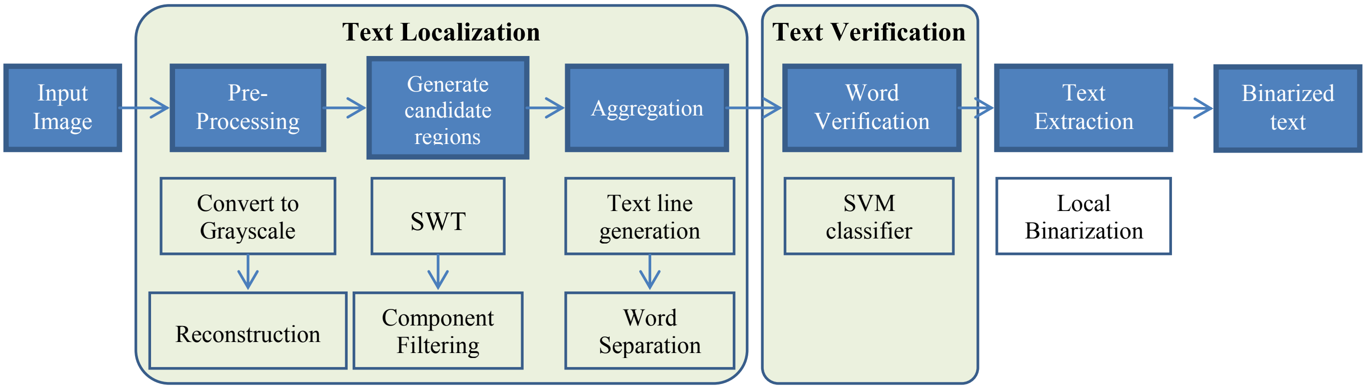
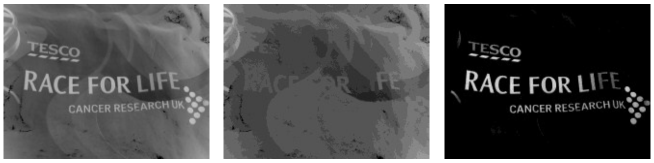


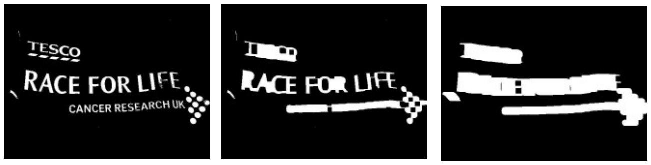
Fig. 1. The flowchart of the proposed text detection and extraction method



(a) (b) (c)

Fig. 2. (a) Gray scale image g, (b) Reconstruction of g from f : $R_g^\delta(f)$, (c) $g - R_g^\delta(f)$

Here, we utilize closing operator with a 13×13 structuring element to the binary image obtained from the previous step to solve this problem (Fig. 3. (b)). A morphological dilation operator can easily connect the very close regions together while leaving those whose positions are far away to each other isolated. In our proposed method, we use a morphological dilation operator with a 33×1 structuring element to get joint areas referred to as text blobs. By applying geometrical constraints such as block height (greater than 8 pixels), aspect ratio (not lower than 0.5), we get candidate text blocks. The region whose size is too small will also be eliminated. The candidate region is shown in Fig. 3. (c).



(a) (b) (c)

Fig. 3. Generation of candidate regions. (a) binary image, (b) closing on a), (c) dilation on (b)

1) Component Extraction

Motivated by Epshtein's work on the Stroke Width Transform (SWT), we apply SWT to form connected components. Epshtein et al. [10] have proved that the stroke width feature is robust to distinguish text from other complex objects that are visually similar to text such as vegetation. The output of the SWT is an image of size equal to the size of the input image where each element contains the width of the stroke associated with the pixel (see [10] for details).

Considering the effectiveness and efficiency, the computation of SWT can be quite expensive if the image has complex content. Thus, we only apply SWT on the detected candidate text blocks. Fig. 4. shows the result of stroke width transform, where a darker color corresponds to a shorter stroke width.

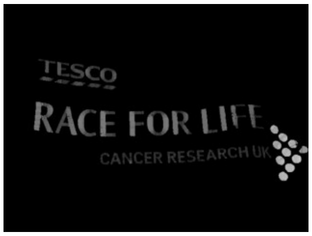


Fig. 4. Stroke width transform result

The next stage is forming connected components. The character components can be generated by merging pixels with similar stroke width value. We do this by considering two pixels as neighboring if the ratio of the stroke widths does not exceed 3.0.

2) Component filtering

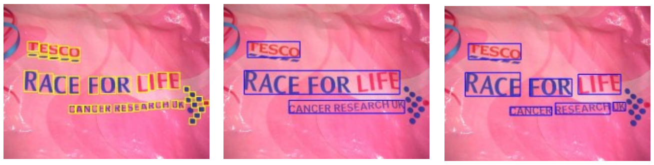
We use some of the heuristic rules to filter the connected components. The most important rule is that we filter components based on the ratio of their stroke width standard deviation (std) to their stroke width mean. The rejection criterion is std/mean > 0.5, which is invariant to scale changes. This threshold was obtained from the training set of the ICDAR competition database. Another important rule is checking if a component's bounding box contains more than three other component's centers, which eliminates signs and frames. Components whose size is too small or too large may be ignored.

C. Group components into word

1) Text line aggregation

The first step of merging consists of grouping adjacent letters in order to form text line. Text lines are important cues for the existence of text, as text almost always appear in the form of straight lines or slight curves. This reasoning allows us to determine whether the connected components belong to text characters or unexpected noises. We first pairwise group the letter candidates using the following rules: i) the ratio of their stroke width medians is lower than 1.5, ii) considering the upper and lower case letters, their height ratio is lower than 2.25, iii) the distance between two letters should not be greater than three times the width of the wider one, iv) the difference between y-coordinates of the connected component centroids should not be greater than 0.5 times the height of the higher one.

Subsequently, text lines are formed based on clusters of pairwise connected letter candidates. We merge together two groups if they share one end and have similar direction. The process is iterated until all text candidates have been assigned to a line, or if there are less than three candidates available within the cluster. A line is declared to be a text line if it contains three or more text objects. The output of this step is illustrated in Fig. 5. (b).



(a) (b) (c)

Fig. 5. (a) Connected components, (b) Text line bounding boxes, (c) Word bounding boxes

2) Word separation

The aim of this step is to separate merged letters into words. To split text lines into words we propose to use the following process which is based on the computation of distances between bounding boxes of letters detected in the previous step (see Fig. 6.). Based on the statistics of the distance distribution (mean and standard deviation values) over the line, a threshold is computed from (2) to split the merged letters in words.

$$T = Mean(D) + \beta \times Std(D) \qquad (2)$$

Threshold T stands for decision whether to split a group of letters or no. We build the distance vector D by measuring the horizontal distances between letters. If the distance between two letters exceeds the threshold, we consider that the two letters belongs to two different words hence they are split. In our system, we set $\beta = 1.5$. The output of this step is illustrated in Fig. 5. (c).

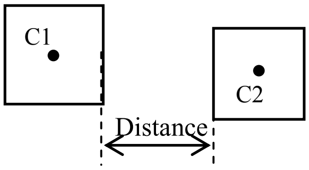


Fig. 6. Distance between two bounding boxes

D. Word Verification

In order to reject falsely detected words, we have developed a verification procedure. We select SVM as classifier in the work. SVM is proposed by Vapnik [13] and have yield excellent results in various binary classification problems in recent years. The important advantages of the support vector classifiers is that it offers a possibility to train generalizable, nonlinear classifiers in high-dimensional spaces using small training set. In this paper, an SVM classifier with the RBF kernel was used.

HOG has been widely used in computer vision. HOG feature shows better performance in characterizing object shape and appearance and it is not sensitive to illumination change. To calculate the HOG feature, the gradient vector of each image pixel within a predefined local region is calculated by using Sobel operator. A histogram is subsequently created by using accumulated gradient vectors, and each histogram-bin is set to correspond to a gradient direction. In order to avoid sensitivity of the HOG feature to illumination, the feature values are often normalized. In this work, we use HOG features to train the SVM classifier for text classification. The SVM was trained on a dataset consisting of 1156 text regions and 800 non-text regions. All training samples were normalized to 48×16 (width×height) as shown in Fig. 7. In the experiment, we are adopted block size is 8×8, cell size is 2×2 and 9 bins for histogram.

For each candidate word, HOG feature is extracted and used by the SVM classifier to verify whether it is a true word.

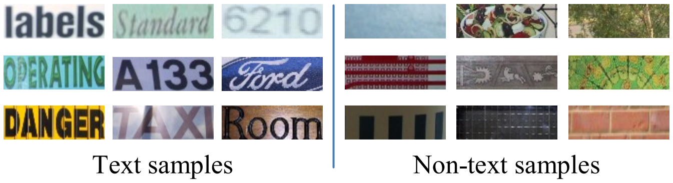


Fig. 7. Some samples from training data

E. Text extraction

Text extraction is an important stage before character recognition. Binarized text regions are used as input to the OCR reading stage. It has been found that global thresholding is not ideal for camera-captured images due to lighting variation. In this paper, we propose a binarization method based on SWT image to extract characters from text line image. The formula to binarize each pixel is defined as follows:

$$b(x) = \begin{cases} 255 & if \ T_1 \le gray(x) \le T_2 \\ 0 & other \end{cases} \qquad (3)$$

where $T_1 = mean(R) - k \times std(R)$ and $T_2 = mean(R) + k \times std(R)$. $mean(R)$ and $std(R)$ are the intensity mean and standard deviation of the pixel whose stroke width value is greater than zero in the text region R. The smoothing term k is set to 1.5 in practical.

F. Text recognition

For text recognition, we apply Google's open source OCR – Tesseract on the binary image. OCR software has many uses pertaining to the processing and archival of printed documents, but its application to photographic images of text still remains challenges. In the paper, we propose a simple OCR post-correction method to improve the accuracy of text recognition in natural scene images. Following the ideas presented in [14], we use Levenshtein distance to choose candidates from the dictionary D selected for correction. The Levenshtein distance between two strings is defined as the minimum number of edits needed to transform one string into the other, with the allowable edit operations being insertion, deletion, or substitution of a single character. Given a word w_{ocr} received from the OCR, we compute the Levenshtein distance between w_{ocr} and each word in the dictionary and retain only those words with the lowest Levenshtein distance as correction candidates. Given word $w = a_1 a_2 \ldots a_n$, a word score $s(w)$ is defined as follows:

$$s(w) = \sqrt[n-2]{\prod_{i=1}^{n-2} P(a_{i+2} \mid a_i a_{i+1})} \qquad (4)$$

The probability $P(a_{i+2} \mid a_i a_{i+1})$ is estimated using relative frequency of the sequence in the dictionary. For each candidate word w_{cand}, we calculate $s(w_{cand})$. The candidate word with the highest score $s(w_{cand})$ is selected.

IV. Indexing and Retrieval

The indexing scheme is quite simple. The images will be automatically annotated by using the words recognized. We allow users to query in two ways: by keywords and by images containing the desired keywords. The difference in query by keywords is to support users to search for images having desired keywords instead of just basing on visual features as the previous query model. In case of being unable to use keywords to query for users not knowing the language of keywords (tourists not knowing the local language), or not supporting devices, we allow users to input images containing queried keywords. Two search modes are supported: exact substring matching and approximate substring matching. Exact substring matching returns all images with substrings in the recognized text that are identical to the search string. Approximate substring matching tolerates a certain number of character differences between the search string and the recognized text. For approximate substring matching, we use the Levenshtein distance between the search string and the text string in images. For each image, we calculate the minimal Levenshtein distance. If the minimal distance is below a certain threshold, the appearance of the string in the image is assumed.

V. Experimental results

A. Text detection and recognition

We preform our experiments on ICDAR 2003 text locating competition dataset [15] which contains TrialTrain and TrialTest sets. This dataset contains images with various resolutions from 307×93 to 1600×1200 pixels, taken both indoor and outdoor. All the text strings in this dataset are in horizontal. The samples to train SVM classifier were collected from 251 TrialTrain set images. All 249 TrialTest set images were used to evaluate the proposed system. The standard definitions of word precision and recall defined in ICDAR 2003 competition were used [15].

TABLE I. Performance comparison of text detection algorithms on ICDAR 2003 dataset.

Method	Precision	Recall	f
Kim's method [17]	0.83	0.62	0.71
Proposed method	**0.78**	**0.62**	**0.69**
Epshtein [10]	0.73	0.60	0.66
Yi [17]	0.67	0.58	0.62
TH-TextLoc [17]	0.67	0.58	0.62
Hinnerk Becker [16]	0.62	0.67	0.62
Neumann [17]	0.69	0.53	0.60

We show the text detection performance on the dataset in TABLE I. The method achieves a recall score similar to the winner of ICDAR 2011 Robust Reading competition [17], but the precision is worse than the ICDAR 2011 winner, which had not been published. However, the proposed method significantly outperforms the second best method of Yi [17] in all three measures. In comparison to previous text detection approaches, our algorithm offers the following major advantages. First, the pre-processing step based on morphological reconstruction helps to exclude substantial none-text objects as well as highlights text ones. Second, HOG feature can capture the texture and structure characteristics of text regions so it is suitable to text detection problem. Further, our system provides a reliable binarization for the detected text. Finally, the proposed algorithm is simple and efficient. Some example results of text detection on the ICDAR 2003 dataset are presented in Fig. 8. Fig. 9. depicts some examples that our method cannot handle to locate the text information.

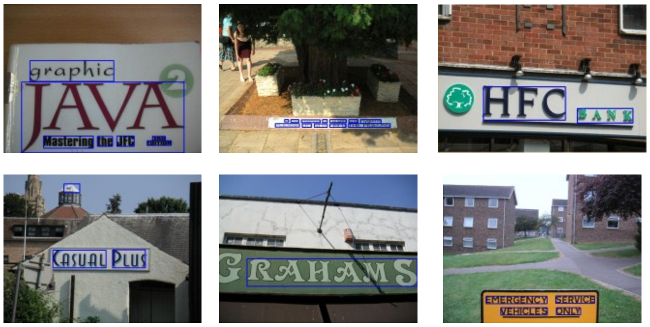


Fig. 8. Some example results of text detection on the ICDAR 2003 dataset

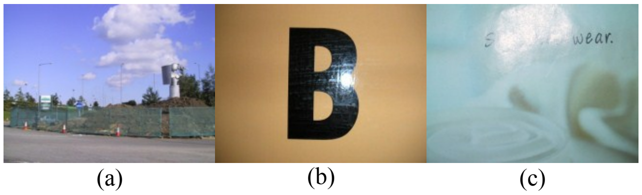
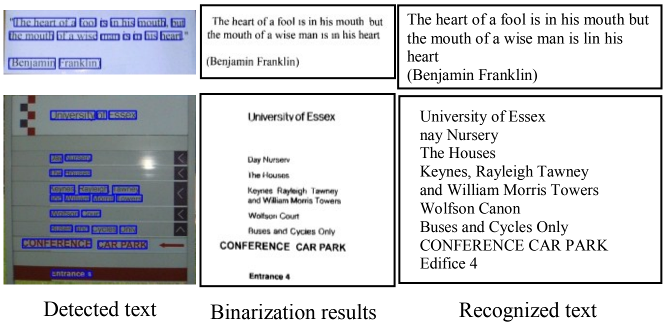


Fig. 9. Examples of images where our method fails: (a) too small size, (b) less than 3 characters, (c) strong highlights

We also evaluated the text recognition preformance on TrialTest set. The performance of the text recognition step is evaluated by two ratio measurements. Precision is the ratio of the number of the words correctly recognized to the total number of the words recognized. Recall is the ratio of the number of the words correctly recognized to the total number of the words detected from the detection stage. TABLE II. shows the recognition performance. Fig. 10. shows examples of text detection and recognition, where the left column is detection result, the middle column is the binarization result and the right column is the recognition result. Experimental results show that the proposed binarization procedure is quite usable.

TABLE II. Recognition performance on the words detected in ICDAR 2003 dataset.

Method	Precision	Recall
Proposed method	0.68	0.68



The heart of a fool is in his mouth but
the mouth of a wise man is lin his
heart
(Benjamin Franklin)

University of Essex
nay Nursery
The Houses
Keynes, Rayleigh Tawney
and William Morris Towers
Wolfson Canon
Buses and Cycles Only
CONFERENCE CAR PARK
Edifice 4

Fig. 10. Binarization results and recognition results by the proposed methods

B. Retrieval effectiveness

In order to evaluate the efficiency of query by text, we use 50 words appearing in image database as query keywords. In order to evaluate the efficiency of query by images, we use 25 images excluded in the database as query image. We use two measures for the evaluation of retrieval effectiveness: precision and recall. Precision specifies the ratio of the number of relevant retrieval results to the total number of returned images. Recall specifies the ratio of the number of relevant results to the total number of relevant images in the image database. We assume that an image depicting the search text is retrieval correctly if at least one word in the search text appears in that image. The result of query by images is shown in Fig. 11. The efficiency of query by keywords and query by images in exact substring matching is shown in TABLE III. Image retrieval model based on scene text has not been proposed and published in any research so far, so we cannot compare the efficiency of the proposed model with other models. Experimental results show that our proposed text detection and recognition algorithms can be effectively used to retrieve relevant images. It shows the encouraging results in querying image only using scene text in images.

VI. Conclusion

In this paper, we have presented an image retrieval method based on scene text. The scene text in images is detected and extracted by the proposed method which is robust against various conditions such as different font style, size, color, and scale of text. The binarized text can be directly used for text recognition purposes. The recognized words are used for indexing and retrieval. Experimental results have shown the proposed text detection method is quite competition to the other best existing methods. Our experimental results also proved that the proposed algorithms are suitable for indexing and retrieval of relevant images from an image database. In future, we will evaluate our system on a large database of images. We also attempt to improve the efficiency and incorporate the algorithms into existing content-based image retrieval systems.

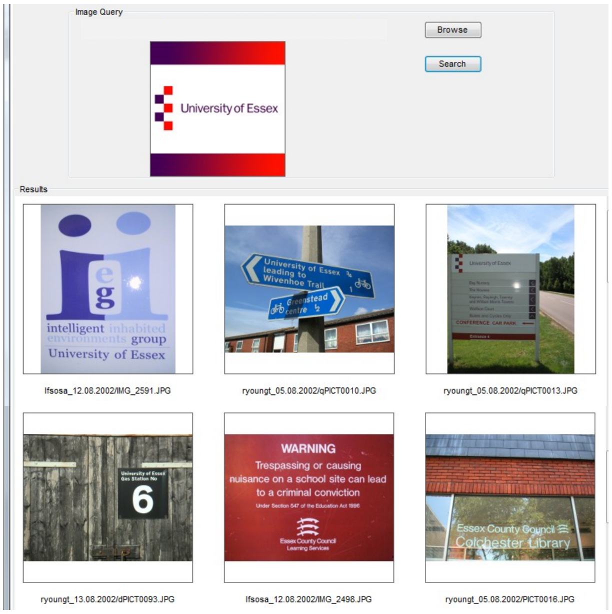


Image Query
Browse
Search
University of Essex
Results
intelligent inhabited environments group University of Essex
lfsosa_12.08.2002/IMG_2591.JPG
University of Essex leading to Wivenhoe Trail
Greenstead Centre
ryoungt_05.08.2002/qPICT0010.JPG
University of Essex
CONFERENCE CAR PARK
ryoungt_05.08.2002/qPICT0013.JPG
University of Essex Gas Station No 6
ryoungt_13.08.2002/dPICT0093.JPG
WARNING
Trespassing or causing nuisance on a school site can lead to a criminal conviction
Essex County Council Learning Services
lfsosa_12.08.2002/IMG_2498.JPG
Essex County Council Colchester Library
ryoungt_05.08.2002/PICT0016.JPG

Fig. 11. Retrieval result by images

TABLE III. Image retrieval performance of the proposed model in exact substring matching

	Precision	Recall
Search by keywords	0.85	0.79
Search by images	0.80	0.64

References

[1] K. Jung, K. I. Kim, and A. K. Jain, “Text in formation extraction in images and video: a survey,” Pattern Recognition, vol. 37, no. 5, pp. 977–997, 2004.

[2] D. T. Chen, J. M. Odobez, and H. Bourland, “Text detection and recognition in images and video frames,” Pattern Recognition, vol. 37, no. 3, pp. 595– 608, 2004.

[3] Q. Ye, W. Gao, W. Wang, W. Zeng, “A robust text detection algorithm in image sand video frames,” Joint Conference of Fourth International Conference on Information Communications and Signal Processing and Pacific-Rim Conference on Multimedia, Singapore 2003.

[4] X. Chen and A. L. Yuille, “Detecting and reading text in natural scenes,” in CVPR, vol. 2, pp. II–366 – II–373, 2004.

[5] S. M. Hanif, L. Prevost, “Text detection and localization in complex scene images using constrained AdaBoost algorithm,” in ICDAR, pp. 1-5, 2009.

[6] Q. Ye, Q. Huang, W. Gao, and D. Zhao, “Fast and robust text detection in images and video frames,” Image Vision Comput., vol. 23, pp. 565–576, 2005.

[7] Z. Liu and S. Sarkar, “Robust outdoor text detection using text intensity and shape features,” in ICPR, pp. 1–4, 2008.

[8] N. Ezaki, M. Bulacu, L. Schomaker, “Text Detection from Natural Scene Images: Towards a System for Visually Impaired Persons,” in ICPR, vol. 2, pp. 683–686, 2004.

[9] R. Datta, D. Joshi, J. Li, and J. Z. Wang, “Image retrieval: Ideas, influences, and trends of the new age,” ACM Computing Surveys, 40(2), 2008.

[10] B. Epshtein, E. Ofek, and Y. Wexler, “Detecting Text in Nature Scenes with Stroke Width Transform,” in CVPR, pp. 2963-2970, 2010.

[11] P. Soille, Morphological Image Analysis: Principles and Applications, pp. 182–198, Springer Verlag, Berlin 2003.

[12] K. Robinson, and P. F. Whelan, Efficient Morphological Reconstruction: A Downhill Filter, Pattern Recognition Letters, Volume 25, Issue 15, pp. 1759–1767, 2004.

[13] V. N. Vapnik, The Nature of Statistical Learning Theory, Springer, 1995.

[14] S. Mihov, S. Koeva, C. Ringlstetter, K. U. Schulz and C. Strohmaier, “Precise and Efficient Text Correction using Levenshtein Automata, Dynamic Web Dictionaries and Optimized Correction Models,” Proceedings of Workshop on International Proofing Tools and Language Technologies, 2004.

[15] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong and R. Young, “ICDAR 2003 Robust Reading Competitions,” in ICDAR, pp. 682-687, 2003.

[16] S. M. Lucas, “ICDAR 2005 text locating competition results,” in ICDAR, vol. 1, pp. 80–84, 2005.

[17] A. Shahab, F. Shafait, and A. Dengel, “ICDAR 2011 robust reading competition challenge 2: Reading text in scene images,” in ICDAR, pp. 1491–1496, 2011.